

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-232803

(43)Date of publication of application : 02.09.1998

(51)Int.Cl.

G06F 12/00  
G06F 9/45

(21)Application number : 09-035822

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 20.02.1997

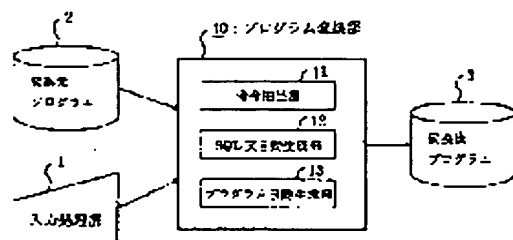
(72)Inventor : KONDO SEIICHI

## (54) PROGRAM CONVERSION SYSTEM

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To provide a program conversion system for easily and surely converting a program sentence by an ISAM form to the program sentence by a relational data base language.

**SOLUTION:** This system is provided with an instruction extraction part 11 for extracting the program sentence relating to access to the data base of the ISAM form from a conversion origin program 2, an automatic SQL sentence generation part 12 for automatically generating an SQL sentence for defining a relational data base and a table, etc., for using data base definition information extracted from the program sentence extracted by the instruction extraction part 11 by applying it to a prepared routine sentence and an automatic program generation part 13 for completing the program sentence required for relational data base access by applying the SQL sentence generated by the automatic SQL sentence generation part 12 to the prepared routine sentence and automatically generating a program (post conversion program 3) for accessing the relational data base by SQL.



## LEGAL STATUS

[Date of request for examination] 20.02.1997

[Date of sending the examiner's decision of rejection] 13.06.2000

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-232803

(43) 公開日 平成10年(1998) 9月2日

(51) Int.Cl.<sup>4</sup>

G 0 6 F 12/00  
9/45

識別記号

5 1 3

F I

G 0 6 F 12/00  
9/44

5 1 3 D  
3 2 2 Z

審査請求 有 請求項の数 8 O L (全 20 頁)

(21) 出願番号

特願平9-35822

(22) 出願日

平成9年(1997) 2月20日

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 近藤 誠一

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

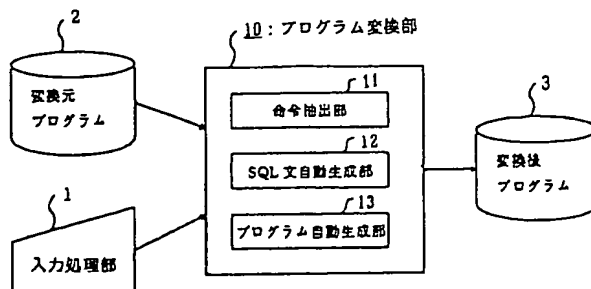
(74) 代理人 弁理士 吉田 研二 (外2名)

(54) 【発明の名称】 プログラム変換システム

(57) 【要約】

【課題】 I S A M形式によるプログラム文を関係データベース言語によるプログラム文に容易かつ確実に変換するプログラム変換システムを提供する。

【解決手段】 変換元プログラム2の中からI S A M形式のデータベースへのアクセスに関連したプログラム文を抽出する命令抽出部11と、命令抽出部11が抽出したプログラム文から抽出したデータベース定義情報を、予め用意した定型文に当てはめることによって使用する関係データベース、テーブル等を定義するSQL文を自動生成するSQL文自動生成部12と、SQL文自動生成部12が生成したSQL文を、予め用意した定型文に当てはめることによって関係データベースアクセスに必要なプログラム文を完成させ、SQLによる関係データベースをアクセスするプログラム(変換後プログラム3)を自動生成するプログラム自動生成部13とを有する。



## 【特許請求の範囲】

【請求項 1】 索引付きアクセスファイルをデータベースとして使用する変換元プログラムに含まれる、使用するデータベース、レコード、データ等を定義する定義文、データベースアクセス命令等の実行文など前記索引付きアクセスファイルアクセスに関連したプログラム文を、関係データベース管理システムによって管理される関係データベースを使用するためのプログラム文に変換することによって関係データベースを使用する変換後プログラムを生成するプログラム変換システムにおいて、前記変換元プログラムの中から索引付きアクセスデータベースアクセスに関連したプログラム文を抽出する文抽出手段と、

前記文抽出手段により抽出された定義文から抽出したデータベース定義情報を予め用意した定型文に当てはめることによって使用する前記関係データベース、テーブルを定義する関係データベース定義文を自動生成する関係データベース定義文自動生成手段と、

前記関係データベース定義文自動生成手段により生成された関係データベース定義文を予め用意した定型文に当てはめることによって前記関係データベースアクセスに必要なプログラムを自動生成する関係データベースプログラム文自動生成手段と、

を有することを特徴とするプログラム変換システム。

【請求項 2】 前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の順ファイルと索引を持つ 0 個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第 1 の定型文を有し、

前記関係データベースプログラム文自動生成手段は、前記第 1 の定型文に対応した第 2 の定型文を有することを特徴とする請求項 1 記載のプログラム変換システム。

【請求項 3】 前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の索引ファイルと索引を持つ 0 個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第 3 の定型文を有し、

前記関係データベースプログラム文自動生成手段は、前記第 3 の定型文に対応した第 4 の定型文を有することを特徴とする請求項 1 記載のプログラム変換システム。

【請求項 4】 前記関係データベース定義文自動生成手段は、前記変換元プログラムがキーでランダムにアクセスすることができる累積型の索引ファイルと索引を持つ 0 個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第 5 の定型文を有し、

前記関係データベースプログラム文自動生成手段は、前記第 5 の定型文に対応した第 6 の定型文を有することを特徴とする請求項 1 記載のプログラム変換システム。

【請求項 5】 前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の順ファイルと索

引を持つ集計ファイルとを使用し、かつ集計命令を記述したプログラム文の実行結果に基づいて前記集計ファイルを更新する形態を有するプログラムである場合に限定使用する第 7 の定型文を有し、

前記関係データベースプログラム文自動生成手段は、前記第 7 の定型文に対応した第 8 の定型文を有することを特徴とする請求項 1 記載のプログラム変換システム。

【請求項 6】 前記関係データベースプログラム文自動生成手段が用意している複数の定型文の中から使用するものを指定する定型文指定手段を有することを特徴とする請求項 1 記載のプログラム変換システム。

【請求項 7】 前記関係データベース定義文自動生成手段は、前記変換元プログラムが前記変換元プログラムが累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を有するプログラムである場合に限定使用され、前記データに基づいて前記各ファイルを結合させる第 9 の定型文を有し、

前記関係データベースプログラム文自動生成手段は、前記第 9 の定型文に対応した第 10 の定型文を有することを特徴とする請求項 1 記載のプログラム変換システム。

【請求項 8】 前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を有するプログラムである場合に限定使用され、前記データに基づいて前記各ファイルを関連付けてビューを生成する第 11 の定型文を有し、

前記関係データベースプログラム文自動生成手段は、前記第 11 の定型文に対応した第 12 の定型文を有することを特徴とする請求項 1 記載のプログラム変換システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 この発明は、情報処理システム上に構築されたデータベースへのアクセスを行うためのプログラムの変換方式、特に I S A M (Indexed Sequential Access Method) 形式のデータベースを使用するプログラムを関係データベースを使用するプログラムに変換する際におけるデータベースアクセス部分の文の変換方式に関する。

【0002】

【従来の技術】 情報処理システムで実行されるプログラムには、システム内に構築されたデータベースへのアクセスを行うための宣言文や実行文等のプログラム文が通常記述されており、プログラム実行時には、この実行文のアクセス結果に基づいて様々な処理が続けて行われる。

【0003】 ところで、システムの利用形態の変更に伴って、情報処理システム上に構築された I S A M 形式のデータベース（以下、「I S A M データベース」）から関係データベースへの移行を行う場合、I S A M データ

ベースと関係データベースでは、アクセス方式が大きく異なるため、データベースを単に入れ換えるだけでなくプログラム内のISAM形式によるプログラム文を関係データベースのプログラム文に新たに書き直す必要が生じてくる。この書換作業は、通常、手動で行われていた。

【0004】しかし、このような手法では、ISAMと関係データベースの両者の知識が不十分であるための実行性能の低下を招き、更には手動による正確さの低下と変換の手間がある。そこで、この問題を解消するものとして、特開昭63-11537号公報に開示された「プログラム変換処理方式」がある。図23は、この従来の「プログラム変換処理方式」を示すブロック図である。この方法は、変換すべきプログラムからデータベースアクセス部分を解析して抽出データを生成し、その情報をもとにプログラム変換を行うものである。

【0005】また、一般的なプログラム変換方法として、特開平1-31233号公報に開示された「プログラミング言語自動変換方法および装置並びに計算機システム」がある。図24は、「プログラミング言語自動変換方法および装置並びに計算機システム」を示すブロック図である。この方法では、1対1に対応しない文法で表現された2つのプログラミング言語の自動変換を、変換対象文の前後関係に基づいて変換効率を上げるものである。

【0006】また、SQL埋め込み分生成に関するものとして、特開平3-182932号公報に開示された「プログラム生成方式」がある。図25は、「プログラム生成方式」を示すブロック図である。この方法では、処理仕様書等を入力としてテーブル定義に基づいてSQLを埋め込んだプログラムを生成するものである。

【0007】

【発明が解決しようとする課題】しかしながら、特開平63-111537号公報に示された従来の「プログラム変換処理方式」では、抽出データごとに処理し、その指令語欄の内容によって変換対応ファイルを索引して、さらにそれに対応する原形プログラムを引くので、ISAMデータベースに対する処理、関係データベースにおけるテーブルへの処理という大幅にその構造が異なるものに対しては、個々の要素の変換のみでは、十分に対応することができないという課題があった。

【0008】また、特開平1-31233号公報に示された従来の「プログラミング言語自動変換方法および装置並びに計算機システム」では、一般の1対1に対応しない文法で表現された2つのプログラミング言語の自動変換に関して対象文の前後を解析して変換後に文を決定する方法を取っているので、特開平63-111537号公報と同様に、局所的な要素の変換のみとなり、ISAMと関係データベースといった大幅に構造が異なる場合には、十分に対応することができないという課題があ

った。

【0009】更に、特開平3-182932号公報に示された従来の「プログラム生成方式」では、処理仕様書等を使用して関係データベースアクセスを目的とするSQLを埋め込んだプログラムを生成することを目的としており、プログラムを変換する過程では、SQL文生成という部分についてのみ利用することが可能となっている。SQL文自動生成に関しては、仕様書から指定するという手段を取っており、SQL文で必要となる情報をすべて仕様書に与えざるを得ないという課題があった。

【0010】本発明は以上のような問題を解決するためになされたものであり、その目的は、データベースを利用するアプリケーションプログラムからISAM形式によるプログラム文を自動的に抽出し、それを関係データベース言語によるプログラム文に容易にかつ確実に変換することができるプログラム変換システムを提供することである。

【0011】

【課題を解決するための手段】以上のような目的を達成するために、第1の発明に係るプログラム変換システムは、索引付きアクセスファイルをデータベースとして使用する変換元プログラムに含まれる、使用するデータベース、レコード、データ等を定義する定義文、データベースアクセス命令等の実行文など前記索引付きアクセスファイルアクセスに関連したプログラム文を、関係データベース管理システムによって管理される関係データベースを使用するためのプログラム文に変換することによって関係データベースを使用する変換後プログラムを生成するプログラム変換システムにおいて、前記変換元プログラムの中から索引付きアクセスデータベースアクセスに関連したプログラム文を抽出する文抽出手段と、前記文抽出手段により抽出された定義文から抽出したデータベース定義情報を予め用意した定型文に当てはめることによって使用する前記関係データベース、テーブルを定義する関係データベース定義文を自動生成する関係データベース定義文自動生成手段と、前記関係データベース定義文自動生成手段により生成された関係データベース定義文を予め用意した定型文に当てはめることによって前記関係データベースアクセスに必要なプログラムを自動生成する関係データベースプログラム文自動生成手段とを有するものである。

【0012】第2の発明に係るプログラム変換システムは、第1の発明において、前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の順ファイルと索引を持つ0個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第1の定型文を有し、前記関係データベースプログラム文自動生成手段は、前記第1の定型文に対応した第2の定型文を有するものである。

【0013】第3の発明に係るプログラム変換システム

は、第1の発明において、前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の索引ファイルと索引を持つ0個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第3の定型文を有し、前記関係データベースプログラム文自動生成手段は、前記第3の定型文に対応した第4の定型文を有するものである。

【0014】第4の発明に係るプログラム変換システムは、第1の発明において、前記関係データベース定義文自動生成手段は、前記変換元プログラムがキーでランダムにアクセスすることができる累積型の索引ファイルと索引を持つ0個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第5の定型文を有し、前記関係データベースプログラム文自動生成手段は、前記第5の定型文に対応した第6の定型文を有するものである。

【0015】第5の発明に係るプログラム変換システムは、第1の発明において、前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の順ファイルと索引を持つ集計ファイルとを使用し、かつ集計命令を記述したプログラム文の実行結果に基づいて前記集計ファイルを更新する形態を有するプログラムである場合に限定使用する第7の定型文を有し、前記関係データベースプログラム文自動生成手段は、前記第7の定型文に対応した第8の定型文を有するものである。

【0016】第6の発明に係るプログラム変換システムは、第1の発明において、前記関係データベースプログラム文自動生成手段が用意している複数の定型文の中から使用するものを指定する定型文指定手段を有するものである。

【0017】第7の発明に係るプログラム変換システムは、第1の発明において、前記関係データベース定義文自動生成手段は、前記変換元プログラムが前記変換元プログラムが累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を有するプログラムである場合に限定使用され、前記データに基づいて前記各ファイルを結合させる第9の定型文を有し、前記関係データベースプログラム文自動生成手段は、前記第9の定型文に対応した第10の定型文を有するものである。

【0018】第8の発明に係るプログラム変換システムは、第1の発明において、前記関係データベース定義文自動生成手段は、前記変換元プログラムが累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を有するプログラムである場合に限定使用され、前記データに基づいて前記各ファイルを関連付けてビューを生成する第11の定型文を有し、前記関係データベースプログラム文自動生成手段は、前記第11の定型文に対応した第12の定型文を有するものである。

【0019】

【発明の実施の形態】以下、図面に基づいて、本発明の

好適な実施の形態について説明する。なお、本実施の形態では、関係データベース言語としてSQLを使用する場合で説明する。もちろん、他の関係データベース言語にも適用可能であることはいうまでもない。

【0020】実施の形態1。図1は、本発明に係るプログラム変換システムの一実施の形態を示した概略構成図である。本システムは、情報処理システムの一部又はその全体を構成する。図1に示した本システムは、命令抽出部11、SQL文自動生成部12及びプログラム自動生成部13を含むプログラム変換部10と、入力処理部1とで構成される。プログラム変換部10は、変換元プログラム2を入力とし、また、必要に応じて入力処理部1からシステムエンジニア等により設定された内容を入力とし、変換後プログラム3に変換するというプログラム変換処理を行うことになる。

【0021】命令抽出部11は、文抽出手段として設けられ、変換元プログラム2の中から索引付きアクセスデータベースアクセスに関連したプログラム文を抽出する。SQL文自動生成部12は、関係データベース定義文自動生成手段として設けられ、命令抽出部11により抽出された定義文から抽出したデータベース定義情報を、予め用意した定型文に当てはめることによって使用する関係データベース、テーブルを定義する関係データベース定義文(SQL文)を自動生成する。プログラム自動生成部13は、関係データベースプログラム文自動生成手段として設けられ、SQL文自動生成部12により生成されたSQL文を、予め用意した定型文に当てはめることによって関係データベースアクセスに必要なプログラム文を完成させ、SQLによる関係データベースをアクセスするプログラム(変換後プログラム3)を自動生成する。

【0022】本実施の形態においては、変換元プログラム2を、ISAMファイルをデータベースとして使用するように記述し、かつ、累積型の順ファイル(累積ファイル)と索引を持つ0個以上のマスターファイルを使用し、累積ファイル“RUISEKI”に含まれるデータに基づいて商品マスターファイル“SHOMAS”を参照するCOBOLプログラムとする。なお、COBOLプログラムの場合、使用する索引付きアクセスファイル、データベース並びにレコード(データ)長、形式等それらに含まれるレコード(データ)に関する情報を事前に定義文で宣言しておき、READ、MOVE等の命令等の実行文でデータベースにアクセスを実際に行うことになる。また、本実施の形態では、データベース内のデータ項目に関する情報を扱うプログラム文(項目移動命令等)も実行文に含まれるものとする。また、本実施の形態においては、変換後プログラム3を、関係データベースをRDO(Remote Data Object)でアクセスするVisual Basicのプログラムとする。

【0023】本実施の形態において特徴的なことは、予

め用意したSQL文に基づく定型文に変換元プログラム2から得たデータベース情報を当てはめることによって関係データベースのプログラム文を自動生成するようにしたことである。これにより、ISAM形式によるプログラムを関係データベースのプログラムに容易にかつ確実に変換することができる。

【0024】次に、本実施の形態における動作について、図2に示したフローチャートを用いて説明する。

【0025】ステップ100において、命令抽出部11は、変換元プログラム2からデータレコード定義(FILE CONTROL, SELECT等)、ファイルアクセス命令(OPEN, READ, MOVE等)、項目移動命令(MOVE)などのプログラム文を抽出する。例えば、図3に示すCOBOLのステートメントが抽出される。なお、この抽出は、予めCOBOLのステートメントリストに含まれるステートメントと変換元プログラム2の各プログラム文とを逐次比較するなど従来からある手法を用いることでできる。

【0026】ステップ110において、SQL文自動生成部12は、上述した累積ファイル及びマスターファイルそれぞれをテーブルに対応させたSQL文を自動生成する。これは、ファイル名をテーブル名に、FD句をSELECT文のフィールドに、キーを入力パラメータに対応させることで生成できる。本実施の形態におけるSQL文自動生成部12は、変換元プログラム2が累積ファイルと索引を持つ0個以上のファイル(マスターファイル)とを使用する形態を有するプログラムである場合に限定使用する図4に示した第1の定型文を有しており、変換元プログラム2から抽出したデータベース定義情報、すなわち累積ファイル、マスターファイル及び各々に含まれるデータの名称をこの定型文の下線部分にはめ込むだけで関係データベース定義文(SQL文)を生成することができる。

【0027】ステップ120において、プログラム自動生成部13は、SQL文自動生成部12が生成したSQL文に基づいて変換後プログラム3として書き込まれるSQL文を自動生成する。本実施の形態では、RUISEKIテーブルから参照された商品IDを示すS-IDが商品マスターを参照するキーとなる。本実施の形態におけるプログラム自動生成部13は、SQL文自動生成部12が持つ第1の定型文に対応した図5に示した第2の定型文を有している。この定型文は、変換後プログラム3の基(枠組み)となる。SQL文自動生成部12が生成したSQL文を図5に示した第2の定型文の下線部分にはめ込むだけで変換後プログラム3を自動生成することができる。

【0028】以上のように、本実施の形態によれば、変換元プログラム2が累積ファイルと索引を持つ0個以上のマスターファイルとを使用する形態を有するプログラムである場合に限定使用する図4に示した第1の定型文を予め用意しておくようにしたので、変換元プログラム

2からは、累積ファイル及びマスターファイルに関する情報の抽出のみに注目すればよいことになる。また、累積ファイルを1レコードずつ読み込み、特定の項目をキーとしてマスターファイルを参照するような構造を予め規定した第2の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。

【0029】実施の形態2. 本実施の形態においては、変換元プログラム2を、ISAMファイルをデータベースとして使用するように記述し、かつ、累積型の索引ファイル(累積ファイル)と索引を持つ0個以上のマスターファイルとを使用し、キー順にソートする形態に合致する累積ファイル“RUISEKI”に含まれるデータに基づいて商品マスターファイル“SHOMAS”を参照するCOBOLプログラムとする。

【0030】次に、本実施の形態における動作について、図6に示したフローチャートを用いて説明する。

【0031】ステップ200において、命令抽出部11は、変換元プログラム2からデータレコード定義等のプログラム文を抽出する。この処理は、上記実施の形態1におけるステップ100と同じなので詳細は省略する。

【0032】ステップ210において、SQL文自動生成部12は、上述した累積ファイル及びマスターファイルそれぞれをテーブルに対応させたSQL文を自動生成する。これは、ファイル名をテーブル名に、FD句をSELECT文のフィールドに、キーを入力パラメータに対応させることができる。本実施の形態におけるSQL文自動生成部12は、変換元プログラム2が累積型の索引ファイル(累積ファイル)と索引を持つ0個以上のファイル(マスターファイル)とを使用する形態を有するプログラムである場合に限定使用する図7に示した第3の定型文を有している。この定型文には、第1の定型文にORDER BY句が付加されており、変換元プログラム2から抽出したデータベース定義情報、すなわち累積ファイル、マスターファイル及び各々に含まれるデータの名称、更には累積ファイルのキーであるRUIIDの順に読み込むようにこの定型文の下線部分にはめ込むだけで関係データベース定義文(SQL文)を生成することができる。

【0033】ステップ220において、プログラム自動生成部13は、SQL文自動生成部12が生成したSQL文に基づいて変換後プログラム3として書き込まれるSQL文を、第3の定型文に対応した第4の定型文を用いることによって自動生成する。本実施の形態では、“RUISEKI”テーブルから参照された商品IDである“S-ID”が商品マスターを参照するキーとなる。変換した結果のVisual Basicのプログラムの内容は、実施の形態1とSQL文の部分異なるだけである。

【0034】以上のように、本実施の形態によれば、変換元プログラム2が累積型の索引ファイルと索引を持つ0個以上のマスターファイルとを使用し、かつ、それら

を関連付けてキー順にソートする形態を有するプログラムである場合に限定使用する図7に示した第3の定型文を予め用意しておくようにしたので、変換元プログラム2からは、累積ファイル及びマスターファイルに関する情報の抽出のみに注目すればよいことになる。すなわち、第3の定型文にキー順読み出しを指定するORDER BY句を設定したので、1レコードずつ読み込み、特定の項目をキーとしてマスターファイルを参照する形態の変換後プログラムへの変換を容易かつ確実に行うことができる。

【0035】実施の形態3. 本実施の形態においては、変換元プログラム2を、ISAMファイルをデータベースとして使用するように記述し、かつ、キーでランダムにアクセスすることができる累積型の索引ファイル(累積ファイル)と索引を持つ0個以上のマスターファイルを使用し、累積ファイル“RUISEKI”に含まれるデータに基づいて商品マスターファイル“SHOMAS”を参照するCOBOLプログラムとする。

【0036】次に、本実施の形態における動作について、図8に示したフローチャートを用いて説明する。

【0037】ステップ300において、命令抽出部11は、変換元プログラム2からデータレコード定義等のプログラム文を抽出する。この処理は、上記実施の形態1におけるステップ100と同じなので詳細は省略する。

【0038】ステップ310において、SQL文自動生成部12は、上述した累積ファイル及びマスターファイルそれぞれをテーブルに対応させたSQL文を自動生成する。これは、ファイル名をテーブル名に、FD句をSELECT文のフィールドに、キーを入力パラメータに対応させることができる。本実施の形態におけるSQL文自動生成部12は、変換元プログラム2がキーでランダムにアクセスすることができる累積型の索引ファイル(累積ファイル)と索引を持つ0個以上のファイル(マスターファイル)とを使用する形態を有するプログラムである場合に限定使用する図9に示した第5の定型文を有している。この定型文には、第1の定型文にWHERE句を付加して累積ファイルのキーである“RUIID”を外部から与えられるように設定されており、この定型文の下線部分にはめ込むだけで関係データベース定義文(SQL文)を生成することができる。

【0039】ステップ320において、プログラム自動生成部13は、SQL文自動生成部12が生成したSQL文に基づいて変換後プログラム3として書き込まれるSQL文を、第5の定型文に対応した第6の定型文を用いることによって自動生成する。本実施の形態では、外部から与えられたキーに基づいて“RUISEKI”テーブルを参照し、その商品IDである“S-ID”が商品マスターを参照するキーとなる。図10に変換した結果のVisual Basicのプログラム(第6の定型文)を示す。図10に示した下線部にSQL文自動生成部12が生成したSQL

L文をはめ込むだけで、変換後プログラム3を自動生成することができる。

【0040】以上のように、本実施の形態によれば、変換元プログラム2がキーでランダムにアクセスすることができる累積型の索引ファイルと索引を持つ0個以上のマスターファイルとを使用するプログラムである場合に限定使用する図9に示した第5の定型文を予め用意しておくようにしたので、変換元プログラム2からは、累積ファイル及びマスターファイルに関する情報の抽出のみに注目すればよいことになる。すなわち、第5の定型文にWHERE句を設定したので、外部から指定したキーに対応するレコードを読み込み、特定の項目をキーとしてマスターファイルを参照する形態の変換後プログラムへの変換を容易かつ確実に行うことができる。

【0041】実施の形態4. 本実施の形態においては、変換元プログラム2を、ISAMファイルをデータベースとして使用するように記述し、かつ、累積型の順ファイル(累積ファイル)と索引を持つ集計ファイルとを使用し、集計命令を記述したプログラム文の実行結果に基づいて集計ファイルを更新する形態を有するCOBOLプログラムとする。より具体的には、累積ファイル“RUISEKI”から売上値“URIAGE”を商品ごとの集計ファイル“SHUUKET”の合計に足し込むプログラムである。

【0042】次に、本実施の形態における動作について、図11に示したフローチャートを用いて説明する。

【0043】ステップ400において、命令抽出部11は、変換元プログラム2からデータレコード定義等のプログラム文を抽出する。例えば、図12に示すCOBOLのステートメントが抽出される。この処理は、上記実施の形態1におけるステップ100と同じなので詳細は省略する。

【0044】ステップ410において、SQL文自動生成部12は、上述した累積ファイル及び集計ファイルそれぞれをテーブルに対応させたSQL文を自動生成する。これは、ファイル名をテーブル名に、FD句をSELECT文のフィールドに、キーを入力パラメータに対応させることができる。本実施の形態におけるSQL文自動生成部12は、変換元プログラム2が累積型の索引ファイル(累積ファイル)と索引を持つファイル(集計ファイル)とを使用し、集計ファイルを更新する形態を有するプログラムである場合に限定使用する図13に示した第7の定型文を有している。従って、この定型文のUPDATE文のSET句においてキーを入力パラメータに対応させる。これにより、合計値を更新する命令を第7の定型文の下線部にはめ込むだけで関係データベース定義文(SQL文)を生成することができる。

【0045】ステップ420において、プログラム自動生成部13は、SQL文自動生成部12が生成したSQL文に基づいて変換後プログラム3として書き込まれるSQL文を、第7の定型文に対応した第8の定型文を用

いることによって自動生成する。本実施の形態では、  
 “RUISEKI” テーブルを順に参照し、その商品IDである  
 “S-ID” に合致する “SHUKEI” テーブルの行の “GOU  
 KEI” フィールドを更新する。図14に変換した結果のV  
 isual Basicのプログラム（第8の定型文）を示す。図  
 14に示した下線部にSQL文自動生成部12が生成し  
 たSQL文をはめ込むだけで、変換後プログラム3を自  
 動生成することができる。

【0046】 以上のように、本実施の形態によれば、変  
 換元プログラム2が累積型の順ファイル（累積ファイ  
 ル）と索引を持つ集計ファイルとを使用し、集計命令を  
 記述したプログラム文の実行結果に基づいて集計ファイ  
 ルを更新するプログラムである場合に限定使用する図1  
 3に示した第7の定型文を予め用意しておくようにした  
 ので、変換元プログラム2からは、累積ファイル及びマ  
 スターファイルに関する情報の抽出のみに注目すればよ  
 いことになる。すなわち、上記第7の定型文を用いるこ  
 とにより累積ファイルを1レコードずつ読み込み、特定  
 の項目をキーとして集計ファイルを参照して、集計デー  
 タを集計ファイルに足し込むという形態の変換後プログ  
 ラムへの変換を容易かつ確実に行うことができる。

【0047】 実施の形態5. 上記各実施の形態において  
 は、予め規定したそれぞれ特有の形態を持つ変換元プロ  
 グラム2を、所定の定型文を用いることによって容易に  
 かつ確実に関係データベースを使用するプログラムへ変  
 換することについて説明した。

【0048】 本実施の形態では、上記実施の形態1～4  
 で示した4種類の形態を例とし、システムエンジニア等  
 がこの中から変換を行う変換元プログラム2に合致する  
 形態を手動で選択できるようにしたことを特徴としてい  
 る。図1に示した入力処理部1は、本実施の形態におい  
 てSQL文自動生成部12が用意している上記複数の定  
 型文の中から使用するものを指定する定型文指定手段と  
 して設けられており、システムエンジニア等が入力処理  
 部1から何れかの形態を選択する。

【0049】 次に、本実施の形態における動作につい  
 て、図15に示したフローチャートを用いて説明する。

【0050】 入力処理部1は、システムエンジニアによ  
 り選択された変換元プログラム2の形態を受け付けると  
 （ステップ500）、そのプログラム形態に応じたいづ  
 れかの変換処理を行うことになる（ステップ510～5  
 40）。すなわち、予め用意している何れかの定型文を  
 使用した変換処理を行うことになる。なお、各変換処理  
 （ステップ510～540）の内容は、上記各実施の形  
 態において説明したので、その内容の詳細は省略する。

【0051】 以上のように、本実施の形態によれば、変  
 換対象とする変換元プログラムの形態と合致するもの  
 を、複数の予め規定した特定の形態からシステムエン  
 ジニア等が手動で選択できるようにしたので、変換元プロ  
 グラムから抽出するデータベース関連の情報を特定する

ことができる。これにより、変換後プログラムの形態を  
 予め規定できるという特徴を持ち、かつ、プログラム変  
 換可能な形態の範囲を拡大することができる。

【0052】 なお、上記説明では、実施の形態1～4で  
 説明した処理のうち何れかを選択するようにしたが、他  
 の定型文の形態を用意した他の変換処理を選択対象とし  
 て追加することができる。

【0053】 実施の形態6. 本実施の形態においては、  
 変換元プログラム2を、累積型の順ファイルのデータに  
 基づいて索引ファイルを参照する形態を有するCOBO  
 Lプログラムとする。特に、本実施の形態では、この累  
 積ファイル “RUISEKI” からマスターファイル “SHOMA  
 S” を参照する部分について、関係データベースの結合  
 処理を含むSQL文を自動生成して変換後プログラム3  
 に埋め込むことを特徴としている。

【0054】 次に、本実施の形態における動作につい  
 て、図16に示したフローチャートを用いて説明する。

【0055】 ステップ600において、命令抽出部11  
 は、変換元プログラム2からデータレコード定義等のプ  
 ログラム文を抽出する。この処理は、上記実施の形態1  
 におけるステップ100と同じであり、図3に示した処  
 理結果を得る。

【0056】 ステップ610において、SQL文自動生  
 成部12は、上述した累積ファイル “RUISEKI” 及びマ  
 スターファイル “SHOMAS” それぞれをテーブルに対応さ  
 せた “RUISEKI” テーブルと “SHOMAS” テーブルとを関  
 係演算の結合処理を含む参照用のSQL文を生成する。  
 本実施の形態におけるSQL文自動生成部12は、変換  
 元プログラム2が累積型の順ファイル（累積ファイル）  
 のデータに基づいて索引ファイルを参照する形態を有す  
 るプログラムである場合に限定使用する第9の定型文を  
 有している。このような変換元プログラム2である場合  
 にこの図17に示した定型文を用いることにより、累積  
 型の順ファイルと索引を持つマスターファイルを参照し  
 て関連付ける形態であるという情報から、 “RUISEKI”  
 テーブルの列 “S-ID” から “SHOMAS” テーブルの列 “SH  
 OID” と関連付けて図17に示した定型文の下線部分に  
 はめ込むだけで関係データベース定義文（SQL文）を  
 自動生成することができる。

【0057】 ステップ620において、プログラム自動  
 生成部13は、SQL文自動生成部12が生成したSQL  
 文に基づいて変換後プログラム3として書き込まれる  
 SQL文を、第9の定型文に対応した第10の定型文を  
 用いることによって自動生成する。図18には、第10  
 の定型文が示されており、この定型文の下線部にSQL  
 文自動生成部12が生成したSQL文をはめ込むこと  
 によって順に行を参照する変換後プログラム3を自動生  
 成することができる。

【0058】 以上のように、本実施の形態によれば、変  
 換元プログラムが累積型の順ファイルのデータに基づい



て索引ファイルを参照する形態を有するプログラムである場合に限定使用する第9の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第9の定型文に対応した変換後プログラムの基になる第10の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。また、累積ファイルからマスターファイルを参照する部分について、関係データベースの結合処理を含むSQL文を自動生成してプログラムの中に埋め込むことにより、プログラム固有のSQL文に基づいて結合されたデータを読み込むだけで、キーを指定してマスターファイルを参照する必要がないため、プログラム構造が単純で実行性能のよい変換後プログラム3を自動生成することができる。

【0059】実施の形態7。本実施の形態においては、変換元プログラム2を、累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を有するCOBOLプログラムとする。特に、本実施の形態では、この累積ファイル“RUISEKI”からマスターファイル“SHOMAS”を参照する部分について、関係データベースの結合処理を含むSQL文を自動生成してビュー“RUISEKI\_SHOMAS”として登録することを特徴としている。

【0060】次に、本実施の形態における動作について、図19に示したフローチャートを用いて説明する。

【0061】ステップ700において、命令抽出部11は、変換元プログラム2からデータレコード定義等のプログラム文を抽出する。この処理は、上記実施の形態1におけるステップ100と同じであり、図3に示した処理結果を得る。

【0062】ステップ710において、SQL文自動生成部12は、上述した累積ファイル“RUISEKI”及びマスターファイル“SHOMAS”それぞれをテーブルに対応させた“RUISEKI”テーブルと“SHOMAS”テーブルとを関係演算の結合処理を含むビュー“RUISEKI\_SHOMAS”生成用のSQL文を自動生成する。本実施の形態におけるSQL文自動生成部12は、変換元プログラム2が累積型の順ファイル（累積ファイル）のデータに基づいて索引ファイルを参照する形態を有するプログラムである場合に限定使用する第11の定型文を有している。このような変換元プログラム2である場合にこの定型文を用いることにより、累積ファイル内のデータに基づいて累積ファイルとマスターファイルとを関連付けてビューを自動生成する。すなわち、累積型の順ファイルと索引を持つマスターファイルを参照して関連付ける形態であるという情報から、“RUISEKI”テーブルの列“S-ID”から“SHOMAS”テーブルの列“SHOID”と関連付けて図20に示した定型文の下線部分にはめ込むだけで関係データベース定義文（SQL文）のビュー定義部分を自動生成することができる。

【0063】更に、SQL文自動生成部12は、ステップ720において、図21に示した第11の定型文の下線部分にはめ込むだけでビューRUISEKI\_SHOMASを参照するSQL文のビュー参照部分を自動生成することができる。

【0064】ステップ730において、プログラム自動生成部13は、SQL文自動生成部12が生成したSQL文に基づいて変換後プログラム3として書き込まれるSQL文を、第11の定型文に対応した第12の定型文を用いることによって自動生成する。図22には、第12の定型文が示されており、この定型文の下線部にSQL文自動生成部12が生成したSQL文をはめ込むことによって“RUISEKI\_SHOMAS”ビューから順に行を参照する変換後プログラム3を自動生成することができる。

【0065】以上のように、本実施の形態によれば、関係データベースをアクセスする変換後プログラム3への変換を容易にかつ確実に行うことができるだけでなく、累積ファイルからマスターファイルを参照する部分について、関係データベースの結合処理を含むSQL文を自動生成してビューとして登録するようにしたので、ビューを読み込むだけでキーを指定してマスターファイルを参照する必要がないため、プログラム構造が単純で実行性能のよい変換後プログラム3を自動生成することができる。

【0066】

【発明の効果】本発明によれば、索引付きアクセスデータベースアクセスに関わる定義文から抽出したデータベース定義情報を予め用意した定型文に当てはめることによって関係データベースを使用する変換後プログラムを生成するようにしたので、そのプログラム変換処理を容易にかつ確実に行うことができる。

【0067】また、変換元プログラムが累積型の順ファイルと索引を持つ0個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第1の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第1の定型文に対応した変換後プログラムの基になる第2の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。

【0068】また、変換元プログラムが累積型の索引ファイルと索引を持つ0個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第3の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第3の定型文に対応した変換後プログラムの基になる第4の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行う

ことができる。

【0069】また、変換元プログラムがキーでランダムにアクセスすることができる累積型の索引ファイルと索引を持つ0個以上のファイルとを使用する形態を有するプログラムである場合に限定使用する第5の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第5の定型文に対応した変換後プログラムの基になる第6の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。

【0070】また、変換元プログラムが累積型の順ファイルと索引を持つ集計ファイルとを使用し、集計命令を記述したプログラム文の実行結果に基づいて集計ファイルを更新するプログラムである場合に限定使用する第7の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第7の定型文に対応した変換後プログラムの基になる第8の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。

【0071】また、変換対象とする変換元プログラムの形態と合致するものを、複数の予め規定した特定の形態からシステムエンジニア等が手動で選択できるようにしたので、変換元プログラムから抽出するデータベース関連の情報を特定することができる。これにより、変換後プログラムの形態を予め規定できるという特徴を持ち、かつ、プログラム変換可能な形態の範囲を拡大することができる。

【0072】また、変換元プログラムが累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を有するプログラムである場合に限定使用する第9の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第9の定型文に対応した変換後プログラムの基になる第10の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。また、特に、累積ファイルから索引ファイルを参照する部分について、関係データベースの結合処理を含む関係データベース定義文を自動生成してプログラムの中に埋め込むことにより、プログラム固有の関係データベース定義文に基づいて結合されたデータを読み込むだけで、キーを指定して索引ファイルを参照する必要がないため、プログラム構造が単純で実行性能のよい変換後プログラムを自動生成することができる。

【0073】また、変換元プログラムが累積型の順ファイルのデータに基づいて索引ファイルを参照する形態を

有するプログラムである場合に、関係データベースの結合処理を含む関係データベース定義文を自動生成してビューとして登録する第11の定型文を予め用意しておくようにしたので、変換元プログラムからは、上記各ファイルに関する情報の抽出のみに注目すればよく、簡便である。また、第11の定型文に対応した変換後プログラムの基になる第12の定型文を予め用意しておくようにしたので、関係データベースをアクセスするプログラムへの変換を容易にかつ確実に行うことができる。また、特に、累積ファイルから索引ファイルを参照する部分について、関係データベースの結合処理を含む関係データベース定義文を自動生成してビューとして登録するようにしたので、ビューを読み込むだけでキーを指定してマスターファイルを参照する必要がないため、プログラム構造が単純で実行性能のよい変換後プログラムを自動生成することができる。

【図面の簡単な説明】

【図1】 本発明に係るプログラム変換システムの一実施の形態を示した概略構成図である。

20 【図2】 実施の形態1における動作を示したフローチャートである。

【図3】 実施の形態1における命令抽出部が抽出した変換元プログラムの一部を示した図である。

【図4】 実施の形態1におけるSQL文自動生成部の出力であるSQL文を示した図である。

【図5】 実施の形態1におけるプログラム自動生成部の出力である変換後プログラムを示した図である。

【図6】 実施の形態2における動作を示したフローチャートである。

30 【図7】 実施の形態2におけるSQL文自動生成部の出力であるSQL文を示した図である。

【図8】 実施の形態3における動作を示したフローチャートである。

【図9】 実施の形態3におけるSQL文自動生成部の出力であるSQL文を示した図である。

【図10】 実施の形態3におけるプログラム自動生成部の出力である変換後プログラムを示した図である。

【図11】 実施の形態4における動作を示したフローチャートである。

40 【図12】 実施の形態4における命令抽出部が抽出した変換元プログラムの一部を示した図である。

【図13】 実施の形態4におけるSQL文自動生成部の出力であるSQL文を示した図である。

【図14】 実施の形態4におけるプログラム自動生成部の出力である変換後プログラムを示した図である。

【図15】 実施の形態5における動作を示したフローチャートである。

【図16】 実施の形態6における動作を示したフローチャートである。

50 【図17】 実施の形態6におけるSQL文自動生成部

の出力であるSQL文を示した図である。

【図18】 実施の形態6におけるプログラム自動生成部の出力である変換後プログラムを示した図である。

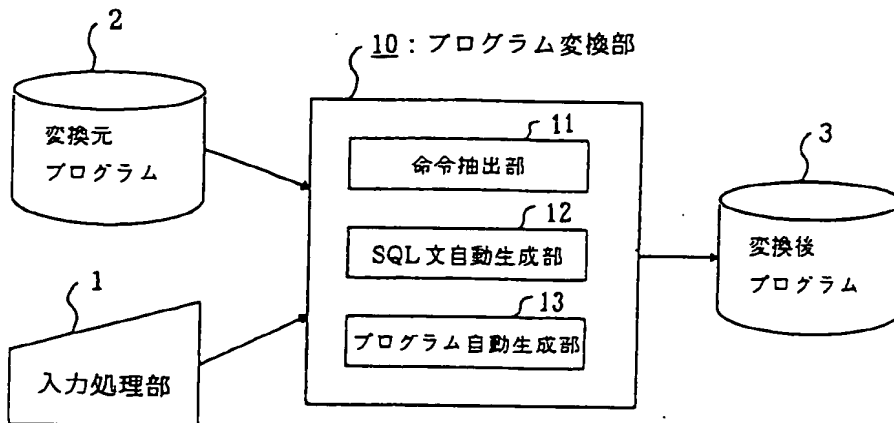
【図19】 実施の形態7における動作を示したフローチャートである。

【図20】 実施の形態7におけるSQL文自動生成部の出力であるSQL文（ビュー定義部分）を示した図である。

【図21】 実施の形態7におけるSQL文自動生成部の出力であるSQL文（ビュー参照部分）を示した図である。

【図22】 実施の形態7におけるプログラム自動生成部の出力である変換後プログラムを示した図である。

【図1】



【図23】 従来の「プログラム変換処理方式」を示すブロック図である。

【図24】 従来の「プログラミング言語自動変換方法および装置並びに計算機システム」を示すブロック図である。

【図25】 従来の「プログラム生成方式」の動作手順を示すフローチャートである。

【符号の説明】

1 入力処理部、2 変換元プログラム、3 変換後プログラム、10 プログラム変換部、11 命令抽出部、12 SQL文自動生成部、13 プログラム自動生成部。

【図4】

```
SELECT RUIID, S-ID, DATA
FROM RUISEKI
```

```
SELECT SHOID, TANKA
FROM SHOMAS
WHERE SHOID = ?
```

【図7】

```
SELECT RUIID, S-ID, DATA
FROM RUISEKI
ORDER BY RUIID
```

```
SELECT SHOID, TANKA
FROM SHOMAS
WHERE SHOID = ?
```

【図9】

```
SELECT RUIID, S-ID, DATA
FROM RUISEKI
WHERE RUIID = ?
```

```
SELECT SHOID, TANKA
FROM SHOMAS
WHERE SHOID = ?
```

【図13】

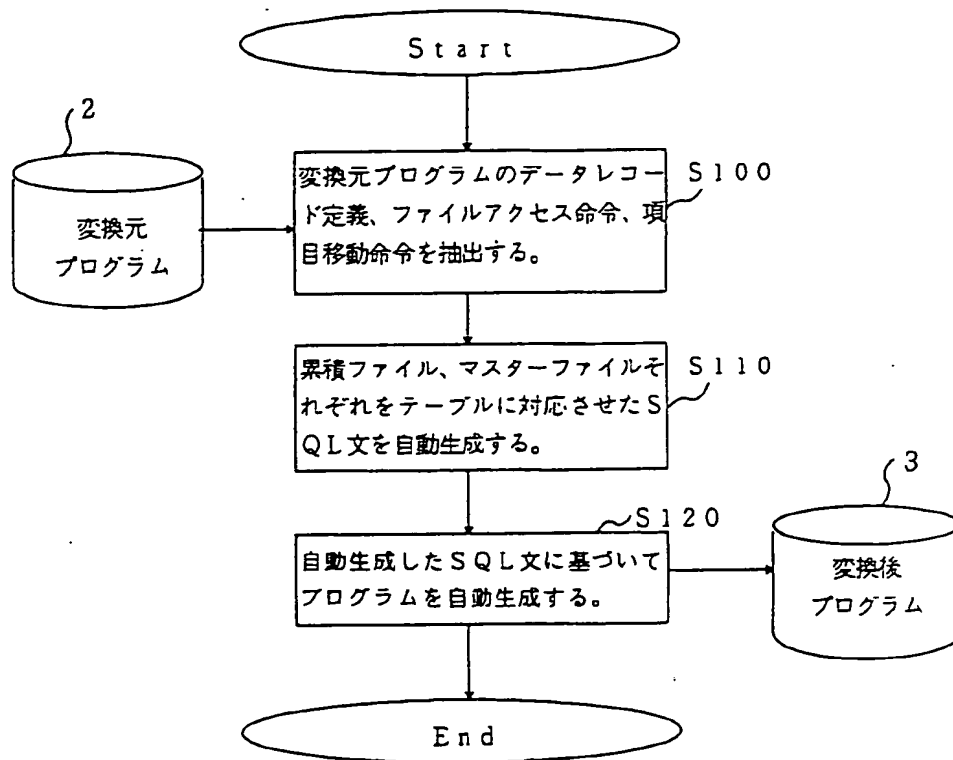
```
SELECT RUIID, S-ID, URIAGE
FROM RUISEKI
```

```
UPDATE SHUUKAI SET GOUKEI=GOUKEI+?
WHERE SHOID = ?
```

【図17】

```
SELECT RUISEKI.RUIID, RUISEKI.S-ID, RUISEKI.DATA, SHOMAS.TANKA
FROM RUISEKI, SHOMAS
WHERE RUISEKI.S-ID = SHOMAS.SHOID
```

【図2】

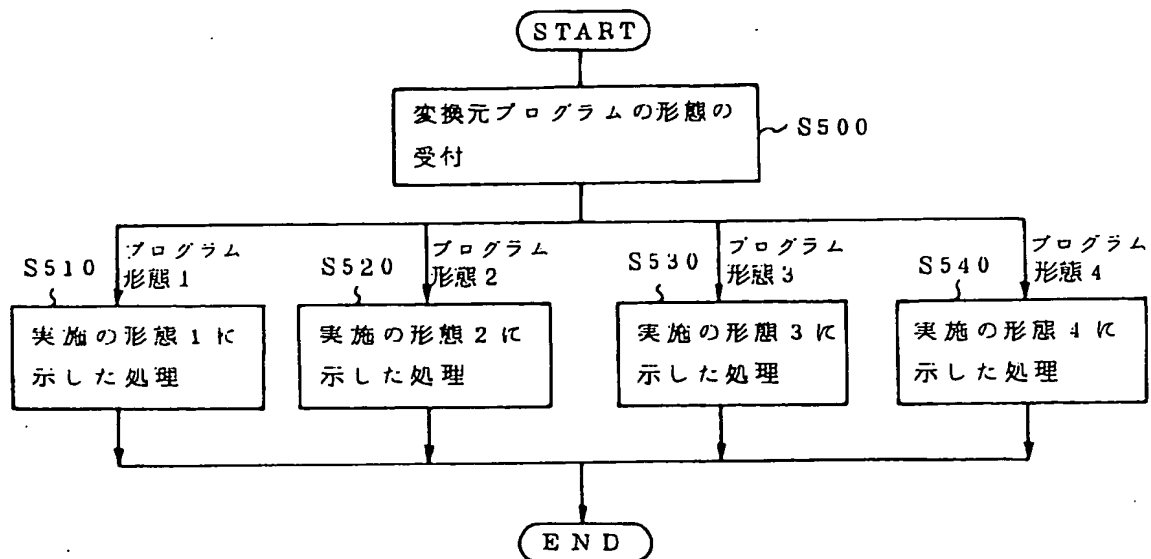


【図21】

```

SELECT *
FROM RUISEKI_SHOMAS
  
```

【図15】



【図3】

## FILE CONTROL.

SELECT RUISEKI ASSIGN TO EXTERNAL RUISEKI  
ACCESS MODE SEQUENTIAL.

SELECT SHOMAS ASSIGN TO EXTERNAL SHOMAS  
ORGANIZATION INDEXED  
ACCESS MODE RANDOM  
RECORD KEY IS SHOID.

## FILE SECTION.

## FD RUISEKI.

01 RUIID            PIC 9(10).  
01 S-ID            PIC 9(10).  
01 DATA           PIC X(100).

## FD SHOMAS.

01 SHOID           PIC 9(10).  
01 TANKA           PIC 9(10).

OPEN INPUT RUISEKI.  
OPEN INPUT SHOMAS.  
READ RUISEKI.  
MOVE S-ID TO SHOID.  
READ SHOMAS.  
CLOSE RUISEKI.  
CLOSE SHOMAS.

【図12】

## FILE CONTROL.

SELECT RUISEKI ASSIGN TO EXTERNAL RUISEKI  
ACCESS MODE SEQUENTIAL.

SELECT SHUUKAI ASSIGN TO EXTERNAL SHUUKAI  
ORGANIZATION INDEXED  
ACCESS MODE RANDOM  
RECORD KEY IS SHOID.

## FILE SECTION.

## FD RUISEKI.

01 RUIID            PIC 9(10).  
01 S-ID            PIC 9(10).  
01 URIAGE          PIC 9(10)  
01 DATA           PIC X(100).

## FD SHUUKAI.

01 SHUUKAI-R  
03 SHOID           PIC 9(10).  
03 GOUKEI          PIC 9(10).

OPEN INPUT RUISEKI.  
OPEN I-O SHUUKAI.  
READ RUISEKI.  
MOVE S-ID TO SHOID.  
READ SHUUKAI.  
ADD URIAGE TO GOUKEI  
REWRITE SHUUKAI-R  
CLOSE RUISEKI.  
CLOSE SHUUKAI.

【図20】

CREATE VIEW RUISEKI\_SHOMAS AS

SELECT RUISEKI.RUIID, RUISEKI.S-ID, RUISEKI.DATA, SHOMAS.TANKA

FROM RUISEKI, SHOMAS

WHERE RUISEKI.S-ID = SHOMAS.SHOID

【図5】

```

Dim En As rdoEnvironment
Dim Cn As rdoConnection,
Dim PsR As rdoPreparedStatement
Dim PsS As rdoPreparedStatement 'マスターファイルの個数分
Dim RsR As rdoResultset
Dim RsS As rdoResultset 'マスターファイルの個数分

Set En = rdoEnvironments(0)
Set Cn = En.OpenConnection("", rdDriverNoPrompt, True, ConnectInfo)
Set PsR = Cn.CreatePreparedStatement("", _
    " SELECT RUIID, S-ID, DATA FROM RUISEKI ")
Set RsR = PsR.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)

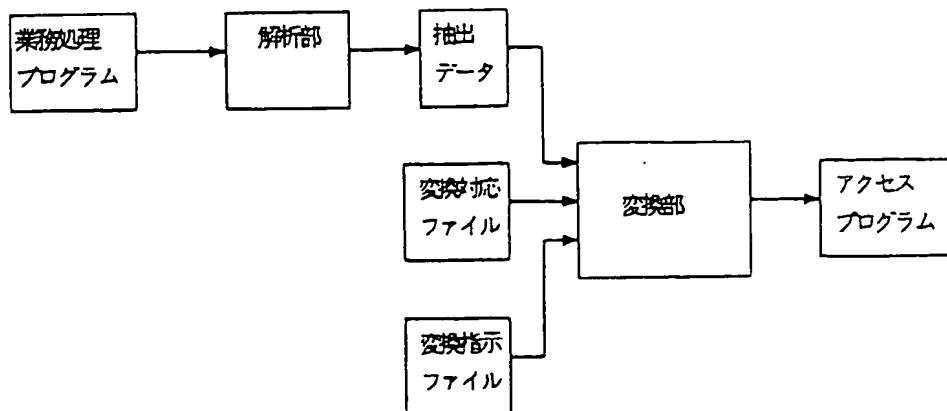
'ここから
Set PsS = Cn.CreatePreparedStatement("", _
    " SELECT SHOID, TANKA FROM SHOMAS WHERE SHOID = ?")
Set RsS = PsS.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)
'ここまで、マスターファイルの個数分

'ここからループの始まり
PsS.rdoParameters(1) = RsR("S-ID")
RsS.Requery
'アプリケーションの処理を記述する。
RsR.MoveNext '次の行に移る
'ここでループが終わる

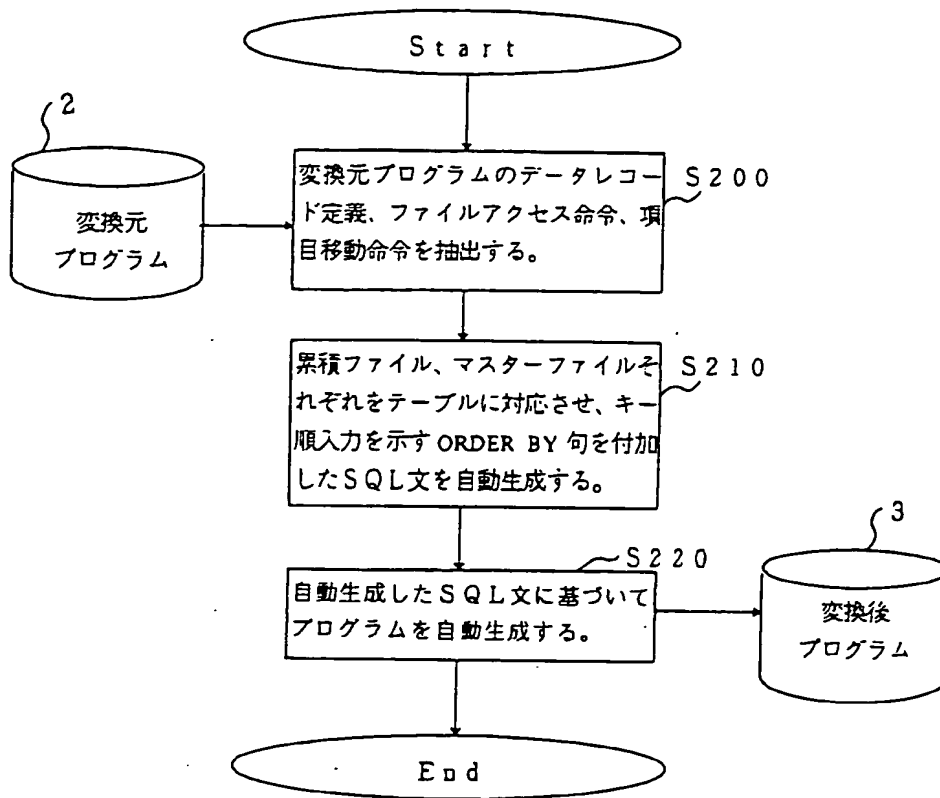
RsR.Close
RsS.Close 'マスターファイルの個数分
PsR.Close
PsS.Close 'マスターファイルの個数分
Cn.Close

```

【図23】



【図6】



【図18】

```

Dim En As rdoEnvironment
Dim Cn As rdoConnection,
Dim RsR As rdoResultset
Dim PsR As rdoPreparedStatement

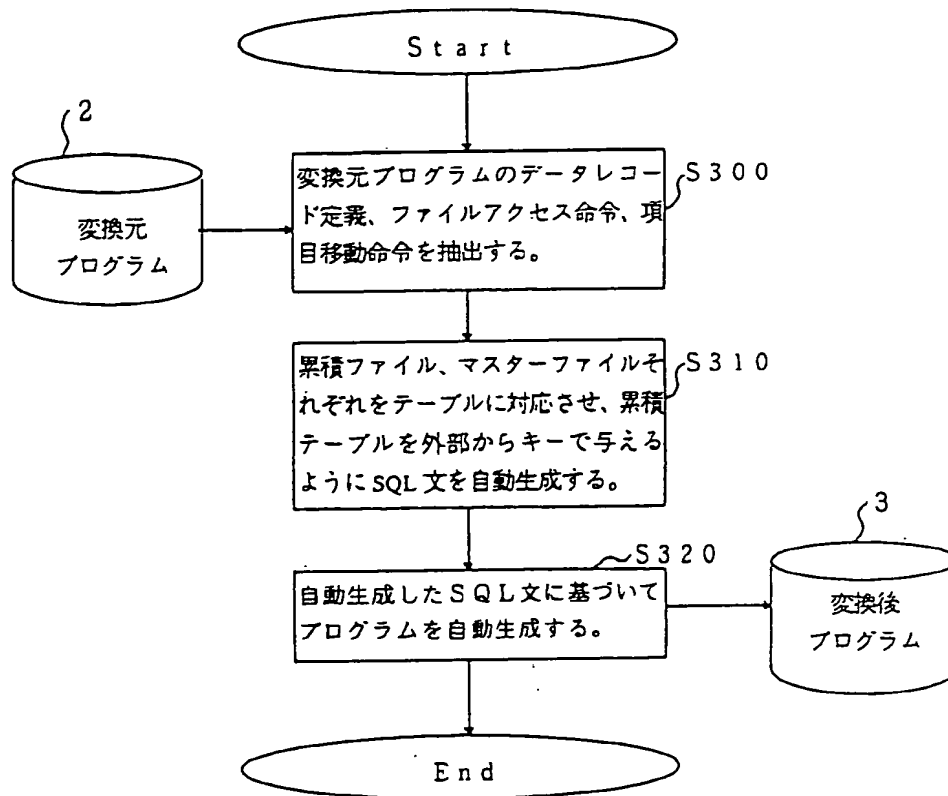
Set En = rdoEnvironments(0)
Set Cn = En.OpenConnection("", rdDriverNoPrompt, True, ConnectInfo)
Set PsR = Cn.CreatePreparedStatement("", _
    "SELECT RUISEKI.RUIID, RUISEKI.S-ID, RUISEKI.DATA, SHOMAS.TANKA" & _
    "FROM RUISEKI, SHOMAS" & _
    "WHERE RUISEKI.S-ID = SHOMAS.SHOID")
Set RsR = PsR.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)
  
```

・ここからループの始まり  
 ・アプリケーションの処理を記述する  
 RsR.MoveNext    ・次の行に移る  
 ・ここでループが終わる

```

RsR.Close
PsR.Close
Cn.Close
  
```

【図8】



【図22】

```

Dim En As rdoEnvironment
Dim Cn As rdoConnection,
Dim PsR As rdoPreparedStatement
Dim RsR As rdoResultset

```

```

Set En = rdoEnvironments(0)
Set Cn = En.OpenConnection("", rdDriverNoPrompt, True, ConnectInfo)
Set PsR = Cn.CreatePreparedStatement("", _
    " SELECT * FROM RUISEKI_SHOMAS")
Set RsR = PsR.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)

```

```

・ ここからループの始まり
・ アプリケーションの処理を記述する
RsR.MoveNext      ・ 次の行に移る
・ ここでループが終わる

```

```

RsR.Close
PsR.Close
Cn.Close

```



【図10】

```

Dim En As rdoEnvironment
Dim Cn As rdoConnection,
Dim PsR As rdoPreparedStatement
Dim PsS As rdoPreparedStatement 'マスターファイルの個数分
Dim RsR As rdoResultset
Dim RsS As rdoResultset 'マスターファイルの個数分

Set En = rdoEnvironments(0)
Set Cn = En.OpenConnection("", rdDriverNoPrompt, True, ConnectInfo)
Set PsR = Cn.CreatePreparedStatement("", _
    " SELECT RUIID, S-ID, DATA FROM RUISEKI WHERE RUIID = ?")
Set RsR = PsR.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)

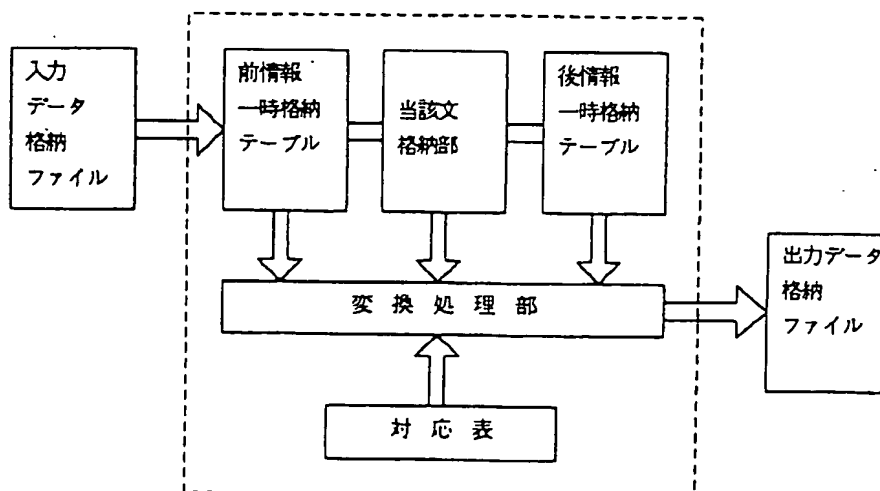
'ここから
Set PsS = Cn.CreatePreparedStatement("", _
    " SELECT SHOID, TANKA FROM SHOWAS WHERE SHOID = ?")
Set RsS = PsS.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)
'ここまで、マスターファイルの個数分

'ここから1回の参照の始まり
PsR.rdoParameters(1) = ' 外部からの入力変数を指定する
RsR.Requery
PsS.rdoParameters(1) = RsR("S-ID")
RsS.Requery
'ここで1回の参照が終わる

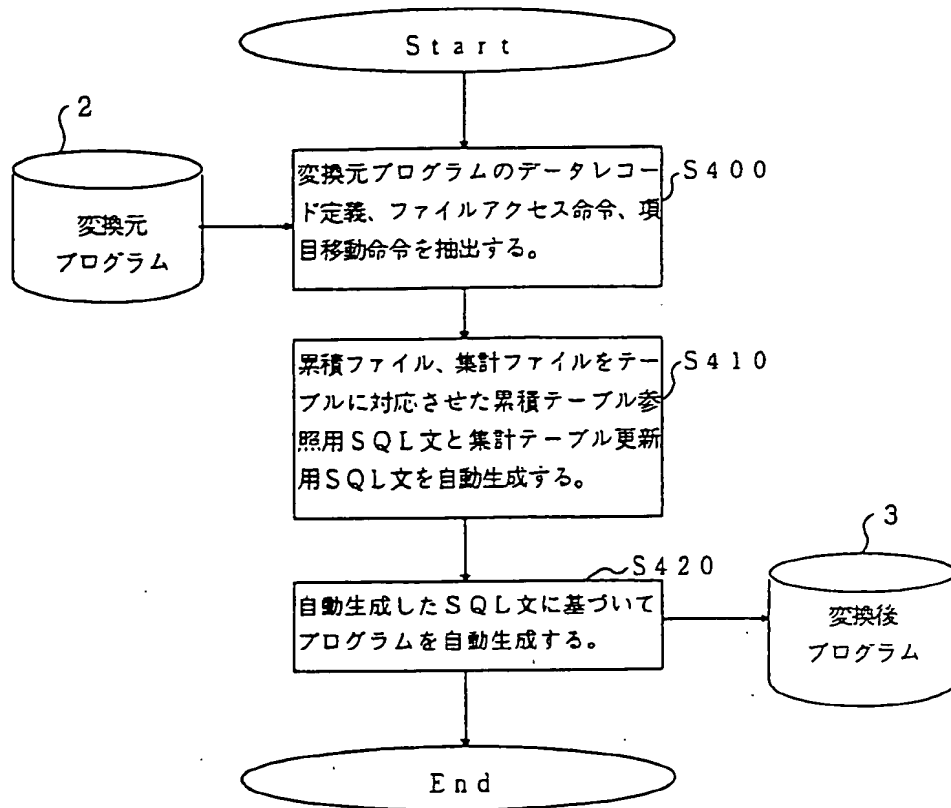
RsR.Close
RsS.Close 'マスターファイルの個数分
PsR.Close
PsS.Close 'マスターファイルの個数分
Cn.Close

```

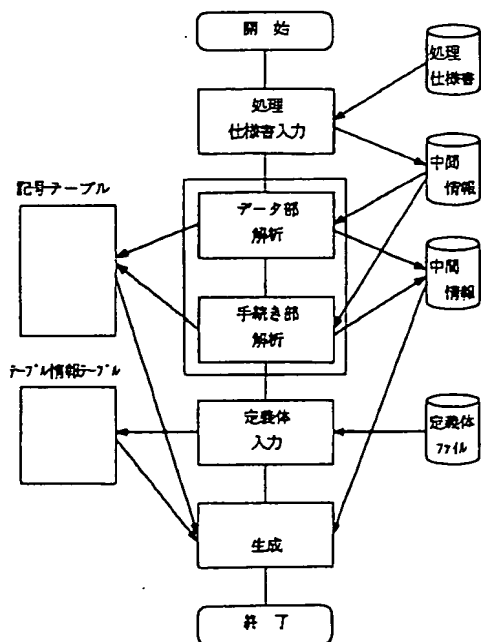
【図24】



【図11】



【図25】



【図14】

```

Dim En As rdoEnvironment
Dim Cn As rdoConnection,
Dim PsR As rdoPreparedStatement
Dim PsS As rdoPreparedStatement
Dim RsR As rdoResultset

Set En = rdoEnvironments(0)
Set Cn = En.OpenConnection("", rdDriverNoPrompt, True, ConnectInfo)
Set PsR = Cn.CreatePreparedStatement("", _
    " SELECT RUIID, S-ID, URIAGE FROM RUISEKI ")
Set RsR = PsR.OpenResultset(rdOpenForwardOnly, rdConcurReadOnly)

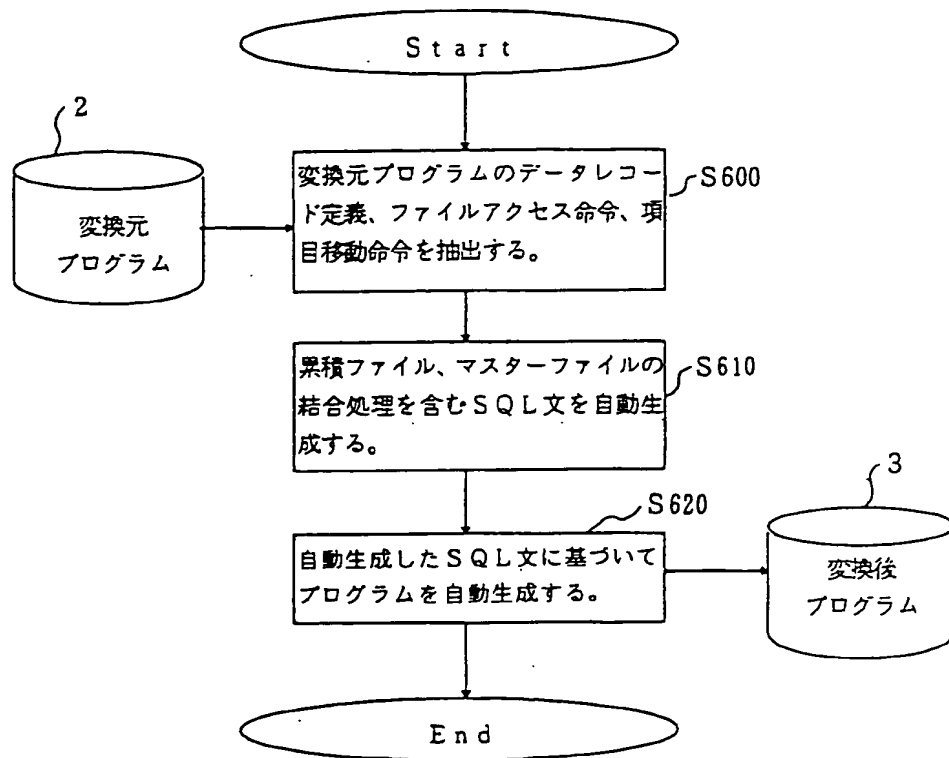
'ここから
Set PsS = Cn.CreatePreparedStatement("", _
    " UPDATE SHUUKAI SET GOUKEI=GOUKEI+? WHERE SHOID = ?")
'ここまで、集計テーブルの準備

'ここからループの始まり
PsS.rdoParameters(1) = RsR("URIAGE")
PsS.rdoParameters(2) = RsR("S-ID")
PsS.Execute
RsR.MoveNext      '次の行に移る
'ここでループが終わる

RsR.Close
PsR.Close
PsS.Close
Cn.Close

```

【図16】



【図19】

